数理解析研究所講究録
1132巻 2000年 1-2

A gold-mining problem — Optimal backup strategy in computer programs

坂口 実 (Minoru Sakaguchi)
神戸商大 濱田年男 (Toshio Hamada)

Abstract

We study an example of R. Bellman's gold-mining problem related to some programming job on computer. The problem is formulated by dynamic programming and the optimal strategy is explicitly derived. The Bayesian version when the parameter involved is unknown is also solved, by the same method. It is shown that the optimal strategy in each of two versions has the "no-island" (or, in other words, "control-limit") property.

1. The Problem

There are n identical items each of which has the probability of failure $p \in (0, 1/2)$. When we use these items to construct a "system" i.e. a series connection of "units," the $\left\{\begin{matrix}1\\2\end{matrix}\right\}$-item unit works "on" with probability $\left\{\begin{matrix}1-p\\1-p^2\end{matrix}\right\}$, and "off" with probability $\left\{\begin{matrix}p\\p^2\end{matrix}\right\}$. We have to choose, one-by-one sequentially, either one of the two kinds of the units, with the objective of

$$\text{E[length of run of "on"-units until END]} \longrightarrow \max. \tag{1.1}$$

subject to

$$2 \times (\text{Number of 2-item units used}) + (\text{Number of 1-item units used}) \leq n \tag{1.2}$$

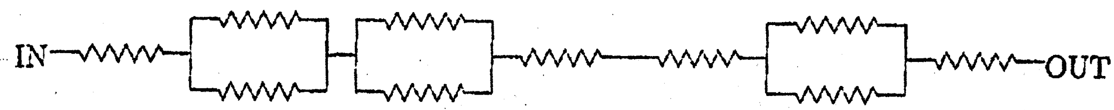

Figure 1: A system with $n = 10$

where END means the event in which, either some unit works off, or there remains no item, whichever occurs first.

Figure 1 shows a system with $n = 10$, and the length of run of "on"-units in this system has the expected value

$$q\left\{p^2 + 2\left(1-p^2\right)p^2 + 3\left(1-p^2\right)^2 p + 4\left(1-p^2\right)^2 qp + \cdots\right\},$$

where $q = 1 - p$.

2. Dynamic Programming

Let F_n be the maximum expected reward when there are n items available. Then the Optimality Equation is evidently

$$F_n = \{q(1+F_{n-1})\} \vee \left\{\left(1-p^2\right)(1+F_{n-2})\right\} \tag{2.1}$$

$$(n = 2, 3, \cdots;\ F_0 = 0,\ F_1 = q)$$

where $q = 1 - p$.

We prove

Theorem 1 *Let $n_0 = [\log 2/(-\log q)]$, and let δ be the strategy: Use 2-item unit as long as $n > n_0$, and switch to 1-item unit as soon as $n \le n_0$. (δ is denoted by $2^{[(n-n_0)/2]}1^{n_0}$.) Then δ is optimal (nearly-optimal) if $n - n_0$ is even (odd). "Nearly-optimal" means that either δ or $2^{[(n-n_0)/2]}1^{n_0+1}$ is optimal.*

3. Bayesian Dynamic Programming

Consider the case where the values of p is unknown. Suppose that there is the prior information that p is a random variable with the distribution beta(α, β), i.e., its pdf is

$$f(p \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1} q^{\beta-1} I(0 < p < 1), \qquad \alpha, \beta \ge 1.$$

Then the Opt. Eq. is

$$F_n(\alpha, \beta) = F_n^1(\alpha, \beta) \vee F_n^2(\alpha, \beta) \tag{3.1}$$

$$(n \geqq 2;\ F_0(\alpha, \beta) = 0,\ F_1(\alpha, \beta) = \beta/(\alpha+\beta))$$

where $F_n^i(\alpha, \beta)$, $i = 1, 2$, is the expected reward in state $(\alpha, \beta \mid n)$ that is obtained when ------.

We prove

Theorem 2 (i) *For the Bayesian Bernoulli/beta version (3.1)~(3.3), there exists a function $n_0(\alpha, \beta)$ such that the optimal strategy in state $(\alpha, \beta \mid n)$ is: Use a $\left\{\begin{matrix} 2 \\ 1 \end{matrix}\right\}$-item unit, if $n \left\{\begin{matrix} > \\ \le \end{matrix}\right\} n_0(\alpha, \beta)$.*

(ii) *$n_0(\alpha, \beta)$ is determined by the positive integer n_0 satisfying*

$$\sum_{k=2}^{n_0-1} k \frac{(\alpha)_2(\beta)_k}{(\alpha+\beta)_{k+2}} + \{(\alpha-1)n_0 - \beta\} \frac{(\beta)_{n_0}}{(\alpha+\beta)_{n_0+1}} < \frac{\alpha\beta}{(\alpha+\beta)_2}$$

$$\le \sum_{k=2}^{n_0} k \frac{(\alpha)_2(\beta)_k}{(\alpha+\beta)_{k+2}} + \{(\alpha-1)(n_0+1) - \beta\} \frac{(\beta)_{n_0+1}}{(\alpha+\beta)_{n_0+2}}, \tag{3.4}$$

where $(m)_k = m(m+1)\cdots(m+k-1)$, and the empty sum is meant by zero.